Amdahl's Law and Applications

Prof. Naga Kandasamy
ECE Department
Drexel University

March 29, 2016

The important principle of computer system design is to make the common case fast, that is, in making a design tradeoff, favor the frequent case over the infrequent case. This principle also applies when determining how to spend resources (e.g., dollars, man power, etc) since the impact on making some occurrence faster is higher if that occurrence is frequent. So, we have to decide what the frequent case is and how much performance can be gained by making that case faster. Amdahl's law is used to quantify this principle.

The performance gain that can be obtained by improving some portion of a computer can be calculated using Amdahl's Law, which states that the performance improvement to be gained from using some faster mode of execution is limited by the fraction of time the faster mode can be used. So, Amdahl's law defines the speedup that can be gained by using a particular feature.

$$\text{Speedup} = \frac{\text{Execution time for the program without using the enhancement}}{\text{Execution time for the program using the enhancement when possible}}$$

Let us say that the speedup (or enhancement) can be used for some fraction of the original program, which we shall term $\text{Fraction}_{\text{enhanced}}$. The speedup achieved when this enhancement is applied is $\text{Speedup}_{\text{enhanced}}$. So, the improved execution time is given by

$$\text{Execution time}_{\text{new}} = \text{Execution time}_{\text{old}} \times \left((1 - \text{Fraction}_{\text{enhanced}}) + \frac{\text{Fraction}_{\text{enhanced}}}{\text{Speedup}_{\text{enhanced}}}\right)$$

The overall speedup is given by

$$\begin{aligned} \text{Speedup} &= \frac{\text{Execution time}_{\text{old}}}{\text{Execution time}_{\text{new}}} \\ &= \frac{1}{(1 - \text{Fraction}_{\text{enhanced}}) + \frac{\text{Fraction}_{\text{enhanced}}}{\text{Speedup}_{\text{enhanced}}}} \end{aligned}$$

Example. Assume that 25% of your program is floating-point operations and that the rest of the program is integer operations. A design team proposes to speed up the floating-point operations by five times using a dedicated floating-point hardware unit. Another team proposes to speed up just

the integer operations by two times. Given that you have the budget to spend on only one design, which design would you fund, and why?

Let us consider the first design proposal involving speeding up the floating-point operations and call it design_A. Since we can speed up 25% of the code using this approach by 5 times, we can apply Amdahl's law to obtain the overall speedup

$$\begin{aligned}\text{Speedup}_\text{A} &= \frac{1}{(1-.25)+\frac{.25}{5}} \\ &= 1.25\end{aligned}$$

Now, consider the second design proposal involving speeding up integer operations and call it design_B. Since we can speed up 75% of the code using this approach by 2 times, the overall speedup is given by

$$\begin{aligned}\text{Speedup}_\text{B} &= \frac{1}{(1-.75)+\frac{.75}{2}} \\ &= 1.6\end{aligned}$$

So, according to Amdahl's law, design_B provides a bigger bang for the buck.

Example: A common transformation used in graphics processors is square root. Suppose FP square root (FPSQR) is responsible for 20% of the execution time of a graphics application. Once proposal is to enhance the FPSQR hardware and speed this operation up by a factor of 10. The other alternative is to just try to make all FP operations on the processor run faster by a factor of 1.6. FP instructions are responsible for half of the execution time of the application. Your design team believes that they can make all FP instructions run faster by 1.6 times with the same effort required for making FPSQR faster. Compare these design alternatives.

We can compare the speedups achieved by the two alternatives.

$$\text{Speedup}_\text{FPSQR} = \frac{1}{(1-0.2)+\frac{0.2}{10}} = \frac{1}{0.82} = 1.22$$

$$\text{Speedup}_\text{FP} = \frac{1}{(1-0.5)+\frac{0.5}{1.6}} = \frac{1}{0.8125} = 1.23$$

Therefore, improving the performance of the FP operations overall is slightly better in terms of speedup.